HP StorageWorks Command View XP

Path Connectivity Command Line Interface (CLI)

This guide provides detailed descriptions of the Path Connectivity CLI commands and step-by-step instructions on how to use each command.

contents

Figures

Tables

This reference guide provides information to help you:

■   Use the Path Connectivity CLI commands.

"About this Guide" topics include:

■   Overview, page 6
■   Conventions, page 7
■   Getting Help, page 9

Overview

This section covers the following topics:

- Intended Audience
- Prerequisites
- Related Documentation

Intended Audience

This book is intended for use by customers and HP authorized service providers who are experienced with the following:

- Disk array hardware and software
- Storage systems

Prerequisites

Before you use the Path Connectivity CLI, make sure you consider the items below.

- Install or upgrade to the most recent version of Command View XP.
- Set up the HP StorageWorks XP disk arrays.
- Install Path Connectivity.
- Refer to the *readme.txt* file on the CD for any last minute announcements.

Related Documentation

In addition to this guide, HP provides corresponding information:

- *HP StorageWorks Command View XP Installation Guide*
- *HP StorageWorks Command View XP Path Connectivity User Guide*
- *HP StorageWorks Command View XP for XP Disk Arrays User Guide*
- *HP StorageWorks Command View XP Command Line Interface (CLI) Reference Guide*
- Command View XP and Path Connectivity online help

Conventions

Conventions consist of the following:

- Document Conventions
- Text Symbols

Document Conventions

The document conventions included in Table 1 apply in most cases.

Table 1: Document Conventions

Element	Convention
Cross-reference links	Blue text: Figure 1
Key and field names, menu items, buttons, and dialog box titles	**Bold**
File names, application names, and text emphasis	*Italics*
User input, command and directory names, and system responses (output and messages)	`Monospace font` `COMMAND NAMES` are uppercase monospace font unless they are case sensitive
Variables	`<monospace, italic font>`
Website addresses	Blue, underlined sans serif font text: http://www.hp.com

Text Symbols

The following symbols may be found in the text of this guide. They have the following meanings.

> **WARNING:** Text set off in this manner indicates that failure to follow directions in the warning could result in bodily harm or death.

> **Caution:** Text set off in this manner indicates that failure to follow directions could result in damage to equipment or data.

Note: Text set off in this manner presents commentary, sidelights, or interesting points of information.

Getting Help

If you still have a question after reading this guide, contact an HP authorized service provider or access our website: http://www.hp.com.

HP Technical Support

Telephone numbers for worldwide technical support are listed on the following HP website: http://www.hp.com/support/. From this website, select the country of origin.

Note: For continuous quality improvement, calls may be recorded or monitored.

Be sure to have the following information available before calling:

- Technical support registration number (if applicable)
- Product serial numbers
- Product model names and numbers
- Applicable error messages
- Operating system type and revision level
- Detailed, specific questions

HP Storage Website

The HP website has the latest information on this product, as well as the latest drivers. Access storage at: http://www.hp.com/country/us/eng/prodserv/storage.html. From this website, select the appropriate product or solution.

HP Authorized Reseller

For the name of your nearest HP authorized reseller:

- In the United States, call 1-800-345-1518
- In Canada, call 1-800-263-5868
- Elsewhere, see the HP website for locations and telephone numbers: http://www.hp.com.

Getting Started

1

The Path Connectivity Command Line Interface (CLI) is a text-based interface used to query for information about connectivity and accessibility between XP disk arrays and hosts. It is a Java program that can be executed from a command window.

The CLI can run on any platform that has network connectivity to a Command View management station, and support for the Java Runtime Environment (JRE).

Operating Requirements

■ Be sure that same version of Command View is running on the Command View management station.

■ Install the Command View CLI and Path Connectivity CLI programs on the client platform. You can download the software from the Command View Support page.

Note: Refer to the *Path Connectivity CLI Installation Note* located under the Command View **Support** tab for detailed installation instructions.

■ Be sure that JRE 1.4.2 (Windows) or JRE/RTE 1.4.1.03 (HP-UX) is installed on the client platform (the system from which you run the CLI). Other operating systems or JRE versions are not supported.

■ The client system's path must contain the Java executable's (*java.exe*) directory.

■ To make logging in easier, modify the *E2ECLI.properties* file on the client system where the CLI is to be executed. See E2ECLI.properties File on page 16.

■ Modify the *AuthorizedClients.dat* file on the Command View management station. Refer to the Path Connectivity CLI Installation Note located under the Command View **Support** tab for detailed installation instructions.

■ The client platform must have network connectivity to the Command View management station. Be sure you can ping the Command View management station from the client platform.

■ Be sure that Path Connectivity host agents are installed on the hosts.

■ Be sure that Path Connectivity data collection is running.

■ Be sure you have a valid Command View user name and password.

Running the CLI

Modes of Execution

There are two modes of Path Connectivity CLI execution: interactive mode and batch mode.

Interactive Mode

Use interactive mode (the default mode) to enter one command at a time. In this mode, the Path Connectivity CLI provides a command prompt. You enter commands interactively until you end the session by typing exit.

To start the CLI in interactive mode:

1. Open a command window.
2. Enter: e2ecli.

You will then be prompted for a user name and password.

After you log in, you will be able to enter commands one at a time until you type exit.

Batch Mode

Use batch mode to execute a batch file containing multiple CLI commands. This is useful for submitting a large number of commands. The session ends after the commands in the batch file have executed.

To start the CLI in batch mode:

1. Open a command window.
2. Enter: e2ecli -f <*filename*>.

   where <*filename*> is the path and name of the batch file to be executed.

You will be prompted for a user name and password.

After you log in, the batch file will execute.

Logging In

By default, when you run the CLI program, a login window like the one shown in Figure 1 is displayed.

Figure 1: Login window

To log in, enter your user name and password and click **OK**.

To run the CLI from a telnet session using a command prompt window, use a command prompt window that supports the X Windows function, such as Reflection X, or use the command line login method that bypasses the login window. A command prompt window that does not support the X Windows function will not display the graphical login window.

Bypassing the Login Window

You can include login information in the startup command using the –p option, in which case the login window will not be displayed. This is useful for processing commands from a batch file, as execution will not be halted by the login window. CLI program startup will fail if the login information is not correct.

For interactive mode, enter:

```
e2ecli -p <user/password>
```

For batch mode, enter:

```
e2ecli -p <user/password> -f <filename>
```

Redirecting Output

For both modes, the default output location is stdout, the screen. However, you can redirect output to a file by means of the -o option.

Batch mode:

```
e2ecli -p <user/password> -f <filename>
-o <outputfilename>
```

Interactive mode:

```
command -o <outputfilename>
```

Using the list array_security_by_port command as an example, the input would be:

```
list array_security_by_port -array_sn 10332
-o myresults.txt
```

Syntax

The general syntax of CLI commands is:

```
command <argumenttype> <argument>
```

command

One of the commands in the CLI.

<argumenttype>

A brief descriptor indicating what the following argument means. For example, **-host** indicates that the next item is a host argument, **-dev_file** indicates that the next items is a device file argument, etc.

<argument>

The variable for the argument type. For example, for a **-host** argument type, the argument would be <hostname>.

Example:

```
list port_sharing -array_sn <array serial number>
-port_nm <port_name>
```

or

```
list port_sharing -array_sn 10553 -port_nm CL1-A
```

Enter argument types and arguments in any order as long as all required parameters are provided. Arguments between "[ ]" are optional and arguments with "..." accept multiple values separated by commas.

Commands are not case-sensitive. Device file names and file system mount points are case-sensitive. Also, do not use any colons when referring to WWNs.

Miscellaneous Options

Unless a command explicitly states otherwise, all commands support the following options for modifying the output format.

-d <*character delimiter*>

Separate each column in the output by the specified delimiter character. For example, if you want to produce tab-delimited columns enter **-d \t**.

-noheadings

Do not include the column headers as part of the output.

-xsl <*url*>

Override the default output format and specify the URL of a different XSLT version 1.0 style sheet.

E2ECLI.properties File

You could explicitly specify four options on program execution: -url, -protocol, -host, and -port. However, we strongly recommend you modify the *E2ECLI.properties* file to set these values.

The parameters contained in the *E2ECLI.properties* file are:

-url <*url*>

The URL of the Command View Device Management servlet. Use the following format when specifying a URL:

http://<hostname>/hpstmgmt/servlet/MarsDM

-protocol <*protocol*>

The protocol to be used for communication with the Command View Device Management server. The default protocol is HTTP.

-host <*host*>

The IP address or DNS name of the Command View management station.

-port <*port*>

The port number to be used to connect to the Command View Device Management servlet.

The *E2ECLI.properties* file should be in the same directory as the Path Connectivity CLI executable. A sample of this file is included with the Path Connectivity CLI components on the product CD.

When the *E2ECLI.properties* file is present, you do not need to specify settings on execution. The *E2ECLI.properties* file settings will override any setting (or the absence of settings) given on the command line during startup.

Here is an example:

```
#E2ECLI.properties: Property file
#for Path Connectivity CLI Application

#SERVLET_URL=http://<hostname>/hpstmgmt/servlet/Mar#sDM
replaces "<hostname>" with the CV server's #hostname OR its IP
address
SERVLET_URL=http://br7.cali.xp.com/hpstmgmt/servlet/MarsDm

#SERVLET_HOST=<hostname> replace "<hostname>" with #the CV
server's hostname OR its IP address
SERVLET_HOST=111.222.333.444

#SERVLET_PORT=<port number> replace "<port number>" #with the CV
server's listening port number
SERVLET_PORT=3000

#SERVLET_PROTOCOL=<protocol> replace <protocol> #with the
protocol to use for communication with the #CV server
SERVLET_PROTOCOL=http

#Trace level to be set for Path Connectivity CLI.
#Can be from 1 to 3:
#    1: Errors
#    2: Warnings
#    3: Information
TRACELEVEL = 3
```

AuthorizedClients.dat File

Modify the *AuthorizedClients.dat* file on the Command View management station as described in the Command View installation instructions.

Ending the CLI

- ■ To terminate the Path Connectivity CLI program in interactive mode, use the `exit` command.
- ■ In batch mode the program terminates after processing all commands from the input file.

CLI Command Descriptions

2

This chapter contains the CLI command descriptions that contain the description, full syntax, and examples of the use of each command available on the CLI interface for Path Connectivity. Each command is described using the following format (if applicable):

- Command name and brief description
- Syntax needed to enable command
- Arguments needed (if any) to further specify the command
- Example(s) to illustrate the command usage

diag topology_by_wwn

This command diagnoses the connection status of Fibre Channel paths between the given HBA port's worldwide name and the disk array port's worldwide name.

Syntax

```
diag topology_by_wwn -hba_wwn <hba_worldwide_wwn>
-port_wwn <port_worldwide_name>
[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]
[-wait <seconds>]
```

Arguments

-hba_wwn *<hba_worldwide_name>*

   HBA port worldwide name (restricts output to specified host HBA port worldwide names).

-port_wwn *<port_worldwide_name>*

   Disk array port worldwide name (restricts output to specified disk array port worldwide names).

[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]

   Path health must be one of the following: CRITICAL, WARNING, OK, INSUFF_DATA. The output is restricted to the specified path health status.

[-wait *<seconds>*]

   Maximum time to wait to diagnose all paths.

Output

Text describing the topology paths and connection status for all Fibre Channel paths between hosts and disk arrays.

Examples

```
diag topology_by_wwn -hba_wwn 50060B000008F18F -port_wwn
40030A000006F16F
```

```
diag topology_by_wwn -hba_wwn 50060B000008F18F -port_wwn
40030A000006F16F -wait 120
```

diag topology_paths

This command diagnoses the connection status of Fibre Channel paths between the hosts and the disk arrays.

Syntax

```
diag topology_paths [-host <host_name/IP_address,...>]
[-array_sn <array_serial_number,...>]
[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]
[-wait <seconds>]
```

Arguments

`[-host <host_name/IP_address,...>]`

Host name or IP address. Restricts output to specified host names or IP addresses.

`[-array_sn <array_serial_number,...>]`

Disk array serial number (restricts output to specified disk array serial numbers).

`[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]`

Path health must be one of the following: `CRITICAL`, `WARNING`, `OK`, `INSUFF_DATA`. The output is restricted to the specified path health status.

`[-wait <seconds>]`

Maximum time to wait to diagnose all paths.

Output

Text describing the topology paths and connection status for all Fibre Channel paths between hosts and disk arrays.

Examples

```
diag topology_paths -host hp00231 -array_sn 10332
diag topology_paths -host hp00231 -wait 120
```

list array_ports_hosts

This command takes a disk array serial number and returns port host information. It displays the port, LUN security settings, and host HBA worldwide names for every port whether or not there are authorized or SAN-connected unauthorized hosts attached to the port.

Syntax

```
list array_ports_hosts -array_sn <array_serial_number>
```

Arguments

```
-array_sn <array_serial_number>
```

Disk array serial number.

Output

Text describing the ports, security status, hosts, WWNs, nicknames assigned to the host HBAs, host group names, and connection and access status of the host HBA ports to the disk array ports.

Example

```
list array_ports_hosts -array_sn 10049
```

list array_security_by_ldev

This command takes a disk array serial number, CU, and LDEV combination and returns disk array security information. It returns the security settings for the LDEV, ports, host groups, LUNs, authorized host worldwide names, and host worldwide names that have a physical path to the associated ports.

Syntax

```
list_array_security_by_ldev
-array_sn <array_serial_number> -cu_id <CU_id>
-ldev_id <LDEV_id>
```

Arguments

```
-array_sn <array_serial_number>
```
    Disk array serial number.

```
-cu_id <CU_id>
```
    CU.

```
-ldev_id <LDEV_id>
```
    LDEV.

Output

Text describing the ports, hosts, LUNs, and access to the LDEV.

Example

```
list_array_security_by_ldev -array_sn 10332 -cu 01 -ldev_id 4A
```

list array_security_by_port

This command takes a disk array serial number/port name combination and returns disk array security information.

Syntax

```
list array_security_by_port
-array_sn <array_serial_number>
-port_nm <array_port_name>
```

Arguments

`-array_sn <array_serial_number>`

Disk array serial number.

`-port_nm <array_port_name>`

Disk array port name.

Output

Text describing the disk array and port security settings, the security settings for a given port, its host groups and authorized hosts, the LUNs and LDEVs, and all hosts that have a physical path to the port.

Example

```
list array_security_by_port -array_sn 10332 -port_nm CL1-A
```

list bcca_all_mirrors

This command returns a list of all volumes that mirror the same data.

Syntax

```
list bcca_all_mirrors -array_sn <array_serial_number>
-cu_id <CU_id> -ldev_id <LDEV_id> [-h]
```

Arguments

`-array_sn <array_serial_number>`

　　Disk array serial number.

`-cu_id <CU_id>`

　　CU.

`-ldev_id <LDEV_id>`

　　LDEV.

`[-h]`

　　The `-h` option lists hosts and device files connected to the volumes.

Output

Text describing all volumes that mirror the same data.

Example

```
list bcca_all_mirrors -array_sn 10332 -cu 01 -ldev_id 17 -h
```

list bcca_mappings

This command returns host device file names mapped to disk array volumes, with high-level BC0, BC1, BC2, and CA information.

Syntax

```
list bcca_mappings -array_sn <array_serial_number>
[-host <host_name/IP_address,...>]
```

Arguments

`-array_sn <array_serial_number>`

Disk array serial number.

`[-host <host_name/IP_address,...>]`

Host name or IP address.

Output

Text describing host device files attached to the specified disk array and CU:LDEV, with high-level BC0, BC1, BC2, and CA information

Example

```
list bcca_mappings -array_sn 10332 -host hp00231
```

list bcca_pair_status

This command returns BC0, BC1, BC2, and CA pair status information details. If you use the -h option, connected hosts/device files are also listed.

Syntax

```
list bcca_pair_status -array_sn <array_serial_number>
-cu_id <CU_id> -ldev_id <LDEV_id> [-h]
```

Arguments

-array_sn *<array_serial_number>*

   Disk array serial number.

-cu_id *<CU_id>*

   CU.

-ldev_id *<LDEV_id>*

   LDEV.

[-h]

   The -h option lists hosts and device files connected to the volumes.

Output

Text describing the details of the pair status between the selected volume and its corresponding BC0, BC1, BC2, and CA pairs.

Example

```
list bcca_pair_status -array_sn 10332 -cu_id 01 -ldev_id 17
```

list device_info

This command provides a mapping between the device files on the host and the associated port and LDEV on the XP disk array. The `-realtime` option invokes Path Connectivity host agent software on the host remotely and stores the latest data in the database before returning the result.

Syntax

```
list device_info -host <host_name/IP_address> [-c] [-i]
[-l] [-m] [-p] [-r] [-t]
[-dev_file <device_file_name,...>] [-realtime]
```

Arguments

`-host <host_name/IP_address>`

Host name or IP address.

`[-c]`

Displays Continuous Access and Business Copy volume designation information. Data fields include: subsystem number, CT group number, CA volume designation, and BC volume designation for each of the three MU numbers.

`[-i]`

Displays identification information. Data fields include: AL-PA, Target ID, LUN ID, Port ID, LDEV number, LUN type (product ID), and Serial number.

`[-l]`

Sorts output by LDEV number.

`[-m]`

Displays model information. Data fields include: Model ID, Emulation type, Size, Serial number, and Firmware revision.

`[-p]`

Sorts output by disk array port name.

`[-r]`

Displays RAID configuration information. Data fields include: ACP pair number, RAID level, RAID group, and Disk mech.

[-t]

   Displays target, LUN, CU, and LDEV IDs in decimal format. The default is
   hexadecimal format.

[-dev_file <*device_file_name*,...>]

   Produces output containing data only for the specified device file names.

[-realtime]

   By default, the data displayed comes from the database. The -realtime
   option will invoke Path Connectivity host agent software on the host remotely
   and store the latest data in the database before returning the result.

Output

Text describing the mapping of the host and its device files to disk array
mechanisms.

Example

```
list device_info -host hp0231 -m -t -dev_file /dev/rdsk/c45t3d4
```

list event_log

This command lets you retrieve logged events from the Command View management station.

Syntax

```
list event_log [-date_from <mm-dd-yyyy[,hh:mm:ss]>]
[-date_to <mm-dd-yyyy[,hh:mm:ss]>]
[-event <event_type,...>] [-msg <"message">] [-t] [-e]
[-m]
```

Arguments

[-date_from *<mm-dd-yyyy*[,*hh:mm:ss*]>]

> The starting date from which logged events will be listed. Starting time is optional. If you omit this argument, event reporting will begin with the earliest log entry.

[-date_to *<mm-dd-yyyy*[,*hh:mm:ss*]>]

> The ending date to which logged events will be listed. Ending time is optional. If you omit this argument, event reporting will end with the latest log entry.

[-event *<event_type,...>*]

> Restricts output to specified event types. Possible values are: info_event, success_event, error_event, and warn_event.

[-msg <"*message*">]

> Restricts output to events containing the specified message text.

[-t]

> Sorts the output on date and time.

[-e]

> Sorts the output on the event type.

[-m]

> Sorts the output on message text.

Output

Text describing logged events meeting the selection criteria.

Examples

This returns the entire event log sorted (by default) on date and time:

```
list event_log
```

This returns all events from 9-24-2002 0:00:00 to the present:

```
list event_log -date_from 09-24-2002
```

This returns all events up to 9-25-2002 at 2:00:00 p.m.:

```
list event_log -date_to 09-25-2002,14:00:00
```

This returns information events occurring between 09-25-02 1:00 a.m. and 09-25-02 3:30 p.m.:

```
list event_log -date_from 09-25-2002,1:00:00 -date_to
09-25-2002,15:30:00 -event info_event
```

This return all events with message text containing "Data Collection":

```
list event_log -msg "Data Collection"
```

list host_dev_physical_map

This command takes a host name or a host name/device file combination and returns LDEV and disk mechanism ("mech") information. If you include a device file name in the command, information for only that device file is returned.

Syntax

```
list host_dev_physical_map
-host <host_name/IP_address>
[-dev_file <device_file_name,...>]
```

Arguments

-host *<host_name/IP_address>*

   Host name or IP address.

[-dev_file <*device_file_name*,...>]

   Device file name. When device file names are specified, output is restricted to just the specified device file names.

Output

Device file name, disk array ID, LDEV, RAID group, RAID level, and disk mech location.

Example

```
list host_dev_physical_map -host hp00231
```

list is_array_accessible

This command determines if a host can access at least one LDEV on a disk array. It returns TRUE if the disk array is accessible from the host.

Syntax

```
list is_array_accessible -host <host_name/IP_address>
-array_sn <array_serial_number>
```

Arguments

`-host <host_name/IP_address>`

Host name or IP address.

`-array_sn <array_serial_number>`

Disk array serial number.

Output

TRUE if a host can access at least one LDEV; FALSE if not.

Example

```
list is_array_accessible -host hp00231 -array_sn 10332
```

list is_ldev_accessible

This command determines if a host HBA can access an LDEV. It returns TRUE if the LDEV is accessible from the HBA's worldwide name.

Syntax

```
list is_ldev_accessible
-hba_wwn <HBA_port_worldwide_name>
-array_sn <array_serial_number> -cu_id <CU_id>
-ldev_id <LDEV_id>
```

Arguments

`-hba_wwn <HBA_port_worldwide_name>`

Host HBA port worldwide name.

`-array_sn <array_serial_number>`

Disk array serial number.

`-cu_id <CU_id>`

CU.

`-ldev_id <LDEV_id>`

LDEV.

Output

TRUE if the host HBA port worldwide name can access the specified LDEV; FALSE if not.

Example

```
list is_ldev_accessible -hba_wwn 50060B000008F18F -array_sn
10332 -cu 01 -ldev_id 17
```

list is_path_to_array

This command determines if a SAN path exists between a host and a disk array. It returns TRUE if there is a path. Note that the host may not be able to access the disk array due to LUN security.

Syntax

```
list is_path_to_array -host <host name/IP_address>
-array_sn <array_serial_number>
```

Arguments

`-host <host name/IP_address>`

Host name or IP address.

`-array_sn <array_serial_number>`

Disk array serial number.

Output

TRUE if a path exists between the host and the disk array; FALSE if not.

Example

```
list is_path_to_array -host hp00231 -array_sn 10332
```

list is_path_to_ldev

This command determines if a path exists between an HBA port and an LDEV. It returns TRUE if there is a path. Note that the HBA port may not be able to access the LDEV due to LUN security.

Syntax

```
list is_path_to_ldev
-hba_wwn <HBA_port_worldwide_name>
-array_sn <array_serial_number> -cu_id <CU_id>
-ldev_id <LDEV_id>
```

Arguments

-hba_wwn *<HBA_port_worldwide_name>*

Host HBA port worldwide name.

-array_sn *<array_serial_number>*

Disk array serial number.

-cu_id *<CU_id>*

CU.

-ldev_id *<LDEV_id>*

LDEV.

Output

TRUE if a path exists; FALSE if a path does not exist.

Example

```
list is_path_to_ldev -hba_wwn 50060B000008F18F -array_sn 10332
-cu_id 01 -ldev_id 17
```

list is_path_to_port

This command determines if a SAN path exists between an HBA port and a disk array port. It returns TRUE if there is a path. Note that the HBA port may not be able to access the disk array port due to LUN security.

Syntax

```
list is_path_to_port
-hba_wwn <HBA_port_worldwide_name>
-array_sn <array_serial_number>
-port_nm <array_port_name>
```

Arguments

-hba_wwn *<HBA_port_worldwide_name>*

    Host HBA port worldwide name.

-array_sn *<array_serial_number>*

    Disk array serial number.

-port_nm *<array_port_name>*

    Disk array port name.

Output

TRUE if a path exists; FALSE if a path does not exist.

Example

```
list is_path_to_port -hba_wwn 50060B000008F18F -array_sn 10332
-port_nm CL1-A
```

list is_port_accessible

This command determines if a host HBA port worldwide name can access a port on a disk array. It returns TRUE if the port is accessible.

Syntax

```
list is_port_accessible
-hba_wwn <HBA_port_worldwide_name>
-array_sn <array_serial_number>
-port_nm <array_port_name>
```

Arguments

-hba_wwn *<HBA_port_worldwide_name>*

HBA port worldwide name.

-array_sn *<array_serial_number>*

Disk array serial number.

-port_nm *<array_port_name>*

Disk array port name.

Output

TRUE if host HBA port worldwide name can access the disk array port; FALSE if not.

Example

```
list is_port_accessible -hba_wwn 50060B000008F18F -array_sn
10332 -port_nm CL1-A
```

list ldev_sharing

This command returns all hosts that are assigned to share the specified CU:LDEV.

Syntax

```
list ldev_sharing -array_sn <array_serial_number>
-cu_id <CU_id> -ldev_id <LDEV_id>
```

Arguments

-array_sn *<array_serial_number>*

  Disk array serial number.

-cu_id *<CU_id>*

  CU.

-ldev_id *<LDEV_id>*

  LDEV.

Output

Text describing the hosts, HBAs' WWN, and LUNs that share the CU:LDEV.

Example

```
list ldev_sharing -array_sn 10332 -cu_id 01 -ldev_id 17
```

list port_sharing

This command returns all hosts that have access to the specified array's port.

Syntax

```
list port_sharing -array_sn <array_serial_number>
-port_nm <array_port_name>
```

Arguments

```
-array_sn <array_serial_number>
```
   Disk array serial number.

```
-port_nm <array_port_name>
```
   Disk array port name.

Output

Text describing the hosts and HBAs' WWN that share the port.

Example

```
list port_sharing -array_sn 10332 -port_nm CL1-A
```

list topology_by_wwn

This command returns path information about paths between the specified HBA port's worldwide name and the specified disk array port's worldwide name.

Syntax

```
list topology_by_wwn
-hba_wwn <hba_port_worldwide_name>
-port_wwn <array_port_worldwide_name>
[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]
```

Arguments

-hba_wwn *<hba_port_worldwide_name>*

　　HBA port worldwide name.

-port_wwn *<array_port_worldwide_name>*

　　Disk array port worldwide name.

[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]

　　Path health must be one of the following: CRITICAL, WARNING, OK, INSUFF_DATA. The output is restricted to the specified path health status.

Output

Text describing the path and connection between the host and disk array worldwide names.

Example

```
list topology_by_wwn -hba_wwn 50060B000008F18F -port_wwn
40030A000006F16F -path_health CRITICAL
```

list topology_fabric

This command is used to produce a detailed listing of the Fibre Channel paths through the switches.

Syntax

```
list topology_fabric <hba_port_worldwide_name>,
<array_port_worldwide_name>
```

Arguments

<hba_port_worldwide_name>

HBA port worldwide name.

<array_port_worldwide_name>

Disk array port worldwide name.

Output

Text describing the topology fabric and connection status for all Fibre Channel paths through the switches.

Examples

```
list topology_fabric 10000000c92b6b5a, 500060e802756710
```

Output

```
PATH_NR, PORT, STATUS, NAME, PORT, STATUS
1, 20, inSync, sw1, 1, inSync
1, 7, inSync, sw2, 15, inSync
2, 20, inSync, sw1, 2, inSync
2, 11, inSync, sw4, 10, inSync
2, 9, inSync, sw2, 15, inSync
3, 20, inSync, sw1, 3, inSync
3, 4, inSync, sw3, 5, inSync
3, 8, inSync, sw2, 15, inSync
4, 20, inSync, sw1, 3, inSync
4, 4, inSync, sw3, 6, inSync
4, 12, inSync, sw4, 10, inSync
4, 9, inSync, sw2, 15, inSync
```

list topology_paths

This command provides the topology paths and connection status for all Fibre Channel paths between the hosts and the disk arrays. You can restrict the output to specified hosts and disk arrays. You can also restrict the output to paths with a particular status.

Syntax

```
list topology_paths [-host <host_name/IP_address,...>]
[-array_sn <array_serial_number,...>]
[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]
```

Arguments

[-host <*host_name*/*IP_address*,...>]

  Host name or IP address. Restricts output to specified host names or IP addresses.

[-array_sn <*array_serial_number*,...>]

  Disk array serial number (restricts output to specified disk array serial numbers).

[-path_health <CRITICAL|WARNING|OK|INSUFF_DATA>]

  Path health must be one of the following: CRITICAL, WARNING, OK, INSUFF_DATA. The output is restricted to the specified path health status.

Output

Text describing the topology paths and connection status for all Fibre Channel paths between hosts and disk arrays.

Examples

```
list topology_paths
list topology_paths -host hp00231
list topology_paths -host hp00231 -path_health CRITICAL
```

load host_data

This command adds a group of host records to the Path Connectivity database by importing host information from a comma separated variable (CSV) file. This can be useful when you need to add information for a large number of hosts to Path Connectivity and do not want to install the host agent software on the hosts.

Syntax

```
load host_data -data <host_data_file>
[-config <config_file>] [-parser <parser_type>]
```

Arguments

-data <*host_data_file*>

The name of a file, in CSV format, containing the host data to import. Enter the full path and file name. Path Connectivity expects the fields in the CSV file to be in the following order: host_domain_name, ipaddress, node_wwn, port_wwn, device_file_name, tid, lun, ldev, cu, serial_number, array_port_name, and emulation_type.

Path Connectivity interprets the first field to be a host domain name, the second field to be an IP address, etc. If you have a CSV file with fields in a different order, you can express the order in a config file (see the next argument).

[-config <*config_file*>]

If the CSV data file has a different field order than Path Connectivity expects, create a text file, such as *config.txt*, to control the field ordering. For example, the configuration file might specify:

host_domain_name=2

ipaddress=1

node_wwn=3

port_wwn=4

device_file_name=5

When specifying the file, enter the full path and file name.

[-parser <*parser_type*>]

Use this option to indicate the type of parser used. If the CSV file was created in Microsoft Excel, enter excel. For all other files, enter default or omit the -parser option. Default works for all CSV files except those created by Microsoft Excel

Output

The Path Connectivity database is updated to include the information supplied by the CSV file.

Example

```
load host_data -data c:\hostdata\HostSummary.csv -config
c:\hostdata\HostFields.txt -parser excel
```

index

A

audience 6
authorized reseller, HP 9

B

batch mode 13
bypassing the login window 14

C

commands 19
conventions
    document 7
    text symbols 7

D

-d option 16
diag topology_by_wwn 20
diag topology_paths 21
document
    conventions 7
    prerequisites 6
    related documentation 6

E

E2ECLI.properties file 16
ending the CLI 18
exit 18

G

getting help 9
getting started 11

H

help, obtaining 9
-host option 17
HP
    authorized reseller 9
    storage website 9
    technical support 9

I

interactive mode 13

L

list array_ports_hosts 22
list array_security_by_ldev 23
list array_security_by_port 24
list bcca_all_mirrors 25
list bcca_mappings 26
list bcca_pair_status 27
list device_info 28
list event_log 30
list host_dev_physical_map 32
list is_array_accessible 33
list is_ldev_accessible 34
list is_path_to_array 35
list is_path_to_ldev 36
list is_path_to_port 37
list is_port_accessible 38
list ldev_sharing 39
list port_sharing 40
list topology_by_wwn 41
list topology_fabric 42
list topology_paths 43

Figures

Tables